Special Issue on Parallel and Distributed Computing

Foreword

Advances in hardware technologies have led to extensive use of sophisticated processors to build multiprocessors and distributed processors for a spectrum of real-time systems such as home appliances, process control systems, flexible manufacturing systems, flight control systems, and tactical control systems. This in turn has led to challenges in programming distributed, parallel and real-time systems. While distributed programming has been based on the paradigm of achieving a common goal from partial knowledge (information), the principle objective of parallel processing has been to achieve better efficiency and throughput. Even though concurrency has been investigated for so many years, it is only recently that a firm formal foundation of real-time programming has been established. A firm mathematical foundation has made it possible to separate the issues of functional and timing correctness of real-time programs. Such a separation in turn has led to a serious investigation of design, methodology and standardization.

The primary purpose of this special issue is to present the (a) issues and principles of specification and verification of real-time distributed programs, (b) issues of reasoning true concurrency, (c) principle of partial information of distributed programming as reflected in the study of epistemology (or the study of knowledge), and (d) issues of building and programming scalable concurrent computers.

The first paper *Modelling real-time systems: Issues and challenges* by R K Shyamasundar and S Ramesh surveys the issues and challenges that lie in the specification, development, and verification of real-time systems. This paper emphasizes mainly the issues of real-time distributed concurrency.

The paper, *An introduction to compositional methods for concurrency and their application to real-time* by J J M Hooman and W P de Roever discusses formal methods to specify and verify concurrent programs with synchronous message passing. The emphasis is mainly on compositional methods, i.e., methods in which the specification of a compound program can be inferred from specifications of its constituents without reference to the internal structure of those parts. Compositionality enables verification during the process of top-down design (the derivation of correct programs) instead of the more familiar a-posteriori verification based on already completed program code. The paper highlights the main principles behind compositionality by discussing transitions from non-compositional methods to compositional methods for concurrent programs.

The paper *Compositional priority specification in real-time distributed systems* by R K Shyamasundar and L Y Liu discusses the issues of compositional specification of prioritized real-time distributed programming languages. The notion of priority is based on the intuition that a low priority action can proceed only if the high priority action cannot proceed due to lack of a handshaking partner at that point of execution. The notion of priority plays a crucial role in the development of predictable real-time

systems. In this paper, the authors discuss issues of compositional specification of priority and describe an approach wherein one can preserve compositionality without unnecessary restrictions between prioritized events (internal or external) and unprioritized events (local or external).

The paper by G Berry, *A hardware implementation of pure Esterel*, provides a hardware implementation of a subset of Esterel. Esterel is one of the highly developed languages of the class of synchronous concurrent languages dedicated to reactive systems. Unlike the languages discussed in the previous papers this language is based on the synchrony hypothesis wherein it is assumed that control and communication do not take any time. In the translation described, each program generates a specific circuit that responds to any input in one clock cycle. It is shown that whenever the source program satisfies some statically verifiable dynamic properties, the circuit is semantically equivalent to the source program. It is of interest to note that the translation described has been effectively implemented on the programmable active memory Perle0 developed by J Vuillemin and his group at Digital Equipment.

The paper *Models and logics for true concurrency* by Kamal Lodaya, Madhavan Mukund, R Ramanujam and P S Thiagarajan first surveys formal models of distributed systems in which concurrency is specified explicitly, in contrast to more traditional approaches where concurrency is represented implicitly as a nondeterministic choice between all possible sequentializations of concurrent actions. In the second half of the paper, the authors develop a family of logics to specify and reason about the behavioural properties of the models described in the first part. The logics defined are extensions of temporal logic with new modalities to directly describe concurrency.

The paper *Levels of knowledge* by Rohit Parikh and Paul Krasucki discusses a realization of levels of knowledge in distributed systems. After introducing various aspects of the study of knowledge and distributed systems, the authors introduce the notion of levels of knowledge and arrive at protocols that precisely realize levels of knowledge of some formulae.

Nalini Venkatasubramanian, Shakuntala Miriyala and Gul Agha in their paper *Scalable concurrent computing* focus on the challenges in building and programming scalable concurrent computers. The paper describes the inadequacy of current models of computing for programming massively parallel computers and discuss three universal models of concurrent computing development from programming-, architecture- and algorithmic-perspectives. It is shown that the models provide a powerful representation for parallel computing and are shown to be quite close. The paper concludes with a discussion of the use of flexible universal programming models, towards building an environment supporting heterogeneous programming languages.

In the last paper, Amitava Datta, R Srikant, G D S Ramkumar and Kamala Krithivasan present efficient parallel algorithms for the maximum empty rectangle problem. In the paper, the authors describe various parallel algorithms for various architectures and arrive at the complexity of the algorithms.

The idea of bringing out a special issue on the topic owes its origin to Professor N Viswanadham. I thank Professors N Viswanadham and R Narasimha for their interest, enthusiasm, and encouragement which made it possible to bring out the special issue. I also thank all the authors for the articles and their cooperation in

attending to various modifications in a timely fashion and the reviewers for giving the feedback within a short time. Last but not least, I thank the staff of the Indian Academy of Sciences for their help in bringing out the special issue.

R K SHYAMASUNDAR
Guest Editor

March 1992